Airline Crew Scheduling with Regularity

Diego Klabjan [*]
Ellis L. Johnson
George L. Nemhauser

Email: diego,ellis.johnson,george.nemhauser@isye.gatech.edu
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205

Eric Gelman
Srini Ramaswamy
Research and Development
Information Services Division
United Airlines

December 5, 2000

Abstract

The airline crew scheduling problem is the problem of assigning crew itineraries to flights. The problem is hard due to a large number of itineraries and the complicated cost function of an itinerary. The classic approach, which uses a set partitioning formulation, only captures the objective function of minimizing cost. We add a second goal of maximizing the repetition or regularity of itineraries over a weekly horizon and we develop models that capture regularity. We present a new algorithm that solves the crew scheduling problem over a weekly horizon. Solutions are reported that improve on existing solutions used by an airline.

The domestic flight schedules of major U.S. airlines consist of many flights that are flown every day of the week and some exceptions that do not repeat daily, e.g. some flights may be flown every day except Saturday. Thus the schedule repeats weekly, but not quite daily and it is necessary to consider a weekly horizon. The *weekly crew scheduling problem* is to find crew itineraries or pairings that partition all the flights in the weekly schedule. Using the traditional approach, we would minimize the weekly crew cost. However it would be unlikely that the resulting pairings could be repeated many times in the weekly horizon unless such constraints were specifically imposed. Thus the solution would lack *regularity*.

[*] Current address: Department of Mechanical and Industrial Engineering, University of Illinois at Urbana-Champaign, Urbana, IL, 61801

Regularity is important with respect to crew (and plane) schedules, since regular solutions are much easier to implement and manage, and, if possible, crews prefer to repeat itineraries. In current practice, airlines attempt to achieve some regularity by solving a two-stage problem, called *daily/exceptions*. First, the crew scheduling problem with the assumption that all flights flown at least 4 consecutive days per week are operated every day is solved as a *daily problem* by adding fictitious flights to the schedule. Then pairings that contain fictitious flights are removed from the solution and all of the real flights in these removed pairings, together with any flights that are operated fewer than 4 days per week, are put into an auxiliary weekly exceptions problem that is solved separately. The auxiliary problem generally has a sparse schedule that requires deadheading and even its optimal solution yields a high weekly cost. Moreover, if there are many removed pairings, the regularity achieved in stage one will be reduced substantially.

The alternative is to impose regularity constraints or penalties directly on a weekly problem. To the best of our knowledge, this has not been attempted before since it was believed that such an approach would not be computationally tractable. Here we present a new model, called the *weekly crew scheduling model with regularity*, that captures both cost and regularity in a weekly schedule. We demonstrate that by using approximations and integer programming as a heuristic, we obtain solutions that improve on current practice with respect to both regularity and cost.

The rest of the paper is organized as follows. In Section 1 we give a high level description of the models and the methodology. In Section 2 we expand this description by presenting the models and the cost structure. In Section 3 we give details on how we solve the large scale IPs resulting from the models. Section 4 reports computational results.

Airline Crew Scheduling

Most US airlines domestic operations are based on a *hub and spoke* network. Airports with high activity are called *hubs* and the low activity airports are called *spokes*.

A *segment* is a nonstop flight. For each segment i, let dep_i/arr_i be the departure/arrival station and let dt_i/at_i be the departure/arrival time. A *duty* is a working day of a crew that consists of a sequence of flights and is subject to FAA and company rules. A connection within a duty is called a *sit connection*. There is a minimum and maximum sit connection time, denoted by $minSit$ and $maxSit$, respectively. The minimum sit connection time requirement can be violated only if the crew follows the plane turn, i.e. they do not change planes. The cost of a duty, measured in minutes, is usually the maximum of three quantities: the flying time, a fraction of the elapsed time, and the duty minimum guaranteed pay.

Crew bases are designated stations where crews are based. A *pairing* is a sequence of duties, starting and ending at a crew base. A connection between two duties is called an *overnight connection* or *layover*. We refer to the time of a layover as the *rest*. Like sit connection times, there is a lower and an upper bound on the rest. We denote by $minRest$ and $maxRest$ the minimum and maximum allowed rest time, respectively.

The cost of a pairing is also the maximum of three quantities: the sum of the duty costs in the pairing, a fraction of the time away from base and a minimum guaranteed pay times the number of duties. Formally, the cost dc_d of a duty d and the cost pc_p of a pairing p are defined as

$$dc_d = \max\{\sum_{i \in d} fl_i, fd \cdot \text{duty elapsed time}, dmg\}, \tag{1}$$

$$pc_p = \max\{\sum_{d \in p} dc_d, fp \cdot \text{time away from base}, pmg \cdot \sum_{d \in p} 1\}, \tag{2}$$

where fl_i is the flying time of segment i, fd and fp are constants that weight the corresponding elapsed time, and *dmg*, *pmg* are minimum guarantees for the duty and pairing pay, respectively. The *excess cost* or *pay-and-credit*, denoted by α_p of a pairing p, is defined as the cost minus the flying time of the pairing, i.e. $\alpha_p = pc_p - \sum_{i \in p} fl_i$. Note that the excess cost is always nonnegative. The *flight time credit* (FTC) of a pairing is the excess cost times 100 divided by the flying time, i.e. it is pay-and-credit expressed as a percentage of total flying time in the pairing. A pairing is also subject to many FAA rules. In all of our data sets there was a limit of 10 segments per duty and 4 duties per pairing.

Traditionally a crew scheduling problem is modeled as the set partitioning problem

$$\min\{\alpha x : Ax = 1, x \text{ binary}\}, \tag{3}$$

where each variable corresponds to a pairing, $a_{ij} = 1$ if segment i is in pairing j and 0 otherwise, and α_j is the pay-and-credit of pairing j.

The problem is difficult since the number of pairings, i.e. columns, can be extremely large. The number of pairings varies from 200,000 for small fleets, to about a billion for medium size fleets and to billions for large fleets. Furthermore since the cost function of a pairing is nonlinear and the legality rules are complex, it is hard to perform delayed column generation, i.e. generating columns only as they are needed in the optimization algorithm. The literature on crew scheduling is abundant. A recent survey can be found in Barnhart et al. (1999).

The Segment Timeline Network

The *segment timeline network* has two distinct nodes for each flight, one for its arrival and the other for its departure. For each flight there is an arc connecting the two nodes. Additionally the network has an arc between the arrival node of a flight and the departure node of a flight if either the connection corresponds to a plane turn or the connection time between the two flights is within $[minSit, maxSit] \cup [minRest, maxRest]$, and the arrival station of the first flight is the same as the departure station of the second flight. We assume that the arcs wrap around in time, e.g. a flight on Sunday can connect to a flight on Monday. Each pairing is a path in the segment timeline network however, due to pairing feasibility rules, a path is not necessarily a pairing.

1 Overview of the Approach

Our goal is to create a solvable model that captures the trade-off between cost and regularity by varying the objective function. With these two objectives in mind a promising way for solving the problem is to partition the legs into groups in such a way that within each group regularity is easily achievable. For example, if all the legs in a group are flown every day of the week, then every pairing covering these legs can be repeated every day in the week. We partition the legs into g-regular groups, $g = 4, \ldots, 7$, where the legs in a g-regular group can be partitioned with pairings that can be repeated g consecutive days of the week. Note that repeating a pairing g days is not related to the number of duties in the pairing, it means that each flight in the pairing is flown on g consecutive days. For example, if pairing p covers flights i and j on Monday, and flight k on Tuesday, then this pairing can be repeated at least 2 consecutive days if i, j are also flown on Tuesday and k is flown on Wednesday.

1.1 Notation and Problem Definition

A *g-regular leg*, $g = 4, 5, 6, 7$, is the set of segments with the same departure and arrival station, with the same local departure and arrival time, such that the segments in the set are operated on g consecutive days, and g is the maximum number with this property. There are g segments in a g-regular leg. In other words, this set is a repetition of segments flown over several consecutive days, which is very common in domestic schedules. All other segments are *irregular legs* but for convenience we call them 1-*regular legs.* We use this terminology even for a segment that might be repeated 2 or 3 times a week because we consider each of these segments individually. For example, if a segment is operated on 5 days among which at most 4 are consecutive days, then this segment yields one 4-regular leg and one 1-regular leg. Specifically, a segment operated on Monday through Thursday and Saturday yields a 4-regular leg representing the operations on Monday through Thursday and one 1-regular leg corresponding to the Saturday segment. We denote by L_g the set of all g-regular legs and if $i \in L_g$, we define $g_i = g$, i.e. $g_i = |i|$. A *regular leg* is an element of $L(4, 7) = \cup_{g=4}^{7} L_g$. Note that only regular legs are considered in daily problems. We use the term *leg* to refer to either a regular leg or a 1-regular leg. For a leg i, we denote by $i_d \in i$ the segment that is operated on day $d \in W$, where $W = \{\mathrm{Mo}, \mathrm{Tu}, \mathrm{We}, \mathrm{Th}, \mathrm{Fr}, \mathrm{Sa}, \mathrm{Su}\}$ is used to denote the days of the week together with modulo addition defined in the usual wrap around fashion, e.g. $\mathrm{Mo} + 1 = \mathrm{Tu}, \mathrm{Sa} + 4 = \mathrm{We}$. Note that each pairing can be represented as a sequence of segments i_d.

A *g-regular pairing* is a pairing that can be repeated on g consecutive days. Specifically, a pairing $p = (i^1_{d(i_1)}, \ldots, i^k_{d(i_k)})$ can be a g-regular pairing if for all $j = 1, \ldots, k$ the segments $i^j_{d(i_j)+h}$ are operated for each $h, 1 \leq h \leq g - 1$. We use the notation $i_d \in p$ to denote $i_d \in \{i^1_{d(i_1)}, \ldots, i^k_{d(i_k)}\}$ and in this case we say that segment i_d is covered by pairing p. Similarly, we say that leg i is *covered* by pairing p and we write $i \in p$ if there is a day $d \in W$ such that $i_d \in p$. Typically d is known from the context.

Example 1. Consider the following schedule.

	Dep.	Arr.	Dep.	Arr.	Days not	
Leg	Sta.	Sta.	Time	Time	Flown	g
1	LAX	ORD	14:05	19:45		7
2	ORD	DCA	20:30	23:15	67	5
3	DCA	LAX	14:10	18:00	6	6
4	LAX	SFO	10:00	11:30		7
5	SFO	ORD	08:00	13:40		7
6	ORD	LAX	11:30	14:15		7
7	ORD	DCA	07:30	10:30		7
8	DCA	SFO	12:00	15:30		7
9	SFO	ORD	07:00	11:30		7
10	DCA	SFO	16:00	19:00	12345 7	1

The next to last column indicates that the second regular leg is not flown on weekends and that the third one is not flown on Saturday. The last 1-regular leg is operated only on Saturday. The schedule consists of 9 regular legs and one irregular leg. There are seven 7-regular legs ($g_1 = g_4 = g_5 = g_6 = g_7 = g_8 = g_9 = 7$), a single 5-regular leg ($g_2 = 5$), and a single 6-regular leg ($g_3 = 6$). The 2 duty pairing $(1_{\text{Mo}}, 2_{\text{Mo}}, 3_{\text{Tu}})$ can be at most a 4-regular pairing since regular leg 3 is not operated on Saturday, whereas the 3- and 2-duty pairings $(4_{\text{Mo}}, 5_{\text{Tu}}, 6_{\text{We}})$ and $(7_{\text{Mo}}, 8_{\text{Mo}}, 9_{\text{Tu}})$ can be g-regular pairings for any $g \leq 7$.

A possible partition of the regular legs is to assign the 7-regular legs $4, 5, 6, 7, 8, 9$ to the 7-regular group and legs $1, 2, 3$ to the 4-regular group. With such an assignment the segments in $S = \{1_{\text{Fr}}, 1_{\text{Sa}}, 1_{\text{Su}}, 2_{\text{Fr}}, 3_{\text{Su}}, 3_{\text{Mo}}, 10_{\text{Sa}}\}$ still need to be partitioned with 1-regular pairings. An alternative partition is to assign legs $4, 5, 6$ to the 7-regular group, legs $1, 2, 3$ to the 4-regular group and legs $7, 8, 9$ to the 6-regular group. In this case the 4-regular pairing $(1_{\text{Mo}}, 2_{\text{Mo}}, 3_{\text{Tu}})$ partitions the 4-regular group, the 7-regular pairing $(4_{\text{Mo}}, 5_{\text{Tu}}, 6_{\text{We}})$ the 7-regular group, the 6-regular pairing $(7_{\text{Su}}, 8_{\text{Su}}, 9_{\text{Mo}})$ partitions the 6-regular group, and 1-regular pairings have to partition the remaining segments consisting of $S \cup \{7_{\text{Sa}}, 8_{\text{Sa}}, 9_{\text{Su}}\}$. The first solution clearly yields higher regularity, however the second solution is likely to produce lower pay-and-credit since there are more opportunities for good pairings in $S \cup \{7_{\text{Sa}}, 8_{\text{Sa}}, 9_{\text{Su}}\}$. □

1.2 The Regularity Model

A model that partitions the legs into groups requires binary variables for each pairing p in the weekly horizon and a variable for each leg assigning this leg to a regular group. However just knowing the group of a leg is not enough since we have to specify exactly which flights are covered by a pairing in this group. In the second assignment of Example 1, the group covered by 1-regular pairings depends on the 6-regular pairing selected to cover legs $7, 8, 9$. The constraints in the model are to assign each regular leg to a regular group, to partition each group by pairings, and to cover the remaining segments of legs with pairings. The trade-off between regularity and cost is controlled by adjusting the pairing cost and the cost of assigning a leg to a group.

Next we present the formulation of the model. We partition the regular legs into regular groups. Specifically, for a g-regular group, $g = 4, 5, 6, 7$, we require that there are g-regular pairings that partition the regular legs in the group. By definition regular legs i from a g-regular group have $g_i \geq g$.

We define variables for each regular leg that assign the regular leg to a regular group and also specify the day on which the regular leg is in the regular group. If a g-regular leg i is assigned to a g'-regular group and $g' < g$, then there are still some segments of i that need to be covered. The 1-*regular group* consists of all such segments and all the 1-regular legs and is partitioned with 1-regular pairings. In the first assignment of Example 1, the 1-regular group consists of the segments in S whereas in the second assignment the 1-regular group is the set $S \cup \{7_{\text{Sa}}, 8_{\text{Sa}}, 9_{\text{Su}}\}$. The number of segments in the 1-regular group is our measure of regularity, the lower the number, the higher the regularity.

Let $S_g, g \in \{1, 4, 5, 6, 7\}$, be the set of all g-regular pairings. Note that to avoid repetition, we specify that all 7-regular pairings start on Monday. The sets S_g are not disjoint, e.g. a 7-regular pairing has 7 copies in S_6, one for each start day of the week.

Let $G(4,6) = \{4,5,6\}$ and for each leg $i \in L_1 \cup L_4 \cup L_5 \cup L_6$, let sd_i be the day of the week such that i is operated on sd_i but not on $sd_i - 1$. For $i \in L_7$ we define $sd_i = \text{Mo}$. Thus $i = \{i_{sd_i}, \ldots, i_{sd_i+g_i-1}\}$. For example, the schedule from Example 1 yields $sd_2 = \text{Mo}$ and $sd_3 = \text{Su}$. For $i \in L(4,7), d \in W, g \in G(4,6) \cup \{7\}$, we define r_{igd} to be the set of all days $\tilde{d}$ such that $i_{\tilde{d}}$ needs to be covered by a 1-regular pairing, if i_d is covered by a g-regular pairing, i.e. $r_{igd} = \{sd_i, \ldots, sd_i + g_i - 1\} \setminus \{d, \ldots, d + g - 1\}$. In Example 1 we have $r_{3,4,\text{Mo}} = \{\text{Fr}, \text{Su}\}, r_{8,5,\text{We}} = \{\text{Mo}, \text{Tu}\}$.

The following binary variables are needed in the model.

Regularity variables For each $i \in L(4,7)$, for each $g, 4 \leq g \leq g_i$, and for each day $d \in \{sd_i, sd_i + 1, \ldots, sd_i + g_i - g\}$, we define a binary variable x_{igd}. If $i \in L_7$ and $g < 7$, then a variable is defined for each $d \in W$. We set $x_{igd} = 1$ if and only if the segment i_d is covered by a pairing $p \in S_g$, i.e. $i_d \in p$.

Example 2. The regularity variables for the schedule given in Example 1 are

$$\begin{aligned}
&x_{i,7,\text{Mo}}, x_{i,6,d}, x_{i,5,d}, x_{i,4,d} && d \in W, i = 1,4,5,6,7,8,9 \\
&x_{2,5,\text{Mo}}, x_{2,4,\text{Mo}}, x_{2,4,\text{Tu}} && \\
&x_{3,6,\text{Su}}, x_{3,5,\text{Su}}, x_{3,5,\text{Mo}}, x_{3,4,\text{Su}}, x_{3,4,\text{Mo}}, x_{3,4,\text{Tu}} \,. &&
\end{aligned}$$

□

Pairing variables Let the pairing variable $y_p = 1, p \in S_7 \cup S_6 \cup S_5 \cup S_4 \cup S_1$, if and only if the pairing p is in the final weekly crew schedule.

For simplicity of notation we define $R \subseteq L(4,7) \times G(4,6) \times W$ to be $R = \{(i,g,d) : i_d, i_{d+1}, \ldots, i_{d+g-1}$ exist in the schedule$\}$ and $Q = R \cup \{(i, 7, \text{Mo}) : i \in L_7\}$. Note that there is a regularity variable for each element in Q and the elements of R correspond to the regularity variables for $g \in G(4,6)$.

The full regularity model is

$$\min \sum_{\substack{g\in G(4,6)\cup\{1,7\}\\ p\in S_g}} g\cdot\alpha_p\cdot y_p + \sum_{(i,g,d)\in Q} \beta_g\cdot x_{igd}$$

$$\sum_{(i,g,d)\in Q} x_{igd} = 1 \qquad i\in L(4,7) \tag{4}$$

$$\sum_{p\in S_7:i\in p} y_p = x_{i,7,\mathrm{Mo}} \qquad i\in L_7 \tag{5}$$

$$\sum_{p\in S_g:i_d\in p} y_p = x_{igd} \qquad (i,g,d)\in R \tag{6}$$

$$\sum_{p\in S_1:i\in p} y_p = 1 \qquad i\in L_1 \tag{7}$$

$$\sum_{p\in S_1:i_d\in p} y_p = \sum_{\substack{g\in G(4,6)\\ \tilde{d}:d\in r_{ig\tilde{d}}\\ (i,g,\tilde{d})\in Q}} x_{ig\tilde{d}} \qquad i\in L(4,7), d\in W, i_d\in i \tag{8}$$

$$y \ \text{binary},\ x \ \text{binary}\,. \tag{9}$$

Note that in (9) x binary can be relaxed to $x\geq 0$ since from (5) and (6) x binary is implied. The objective function is discussed in Section 2.3. Here it suffices to note that the first term is the usual pairing cost and that the second term is a regularity penalty. The generalized upper bound constraints (4) require that each regular leg be assigned to exactly one regular group. Each regular group has to be partitioned with pairings as specified in constraints (5) and (6). Constraints (7) and (8) yield the same property for the 1-regular group. It is easy to see that we need to cover $i_d, i\in L(4,7)$, in the 1-regular group if and only if

$$\sum_{\substack{g\in G(4,6)\\ \tilde{d}:d\in r_{ig\tilde{d}}\\ (i,g,\tilde{d})\in Q}} x_{ig\tilde{d}} = 1\,.$$

Hence there is no need for regularity variables to be defined for the 1-regular group.

A 7-regular pairing can be assigned by the model to cover, for example, the 5-regular group. Therefore this model is difficult to solve because of the large number of pairing variables, which can be billions even for small size fleets, and in addition can have 10,000 rows for large fleets. The model requires that each regular leg is assigned to a regular group. We make this requirement to control regularity further and to reduce the size of the model, but it can be dropped by considering $G(2,6)=\{2,3,4,5,6\}$ instead of $G(4,6)$. The decomposition approach of first solving the daily problem on all the 7-regular legs does not work since the 7-regular legs are usually not balanced and therefore the problem is infeasible.

1.3 Key Ideas of the Approach

In order to reduce the size of the model, we resort to an approximate model that does not use all of the pairings. Since most legs are flown every day of the week, we anticipate that the 7-regular group will have the largest flying time and therefore we want to have low pay-and-credit in this group. We require that pairings partition the legs in the 7-regular group but for the remaining groups we only require that paths in the segment timeline network partition the legs. Modeling paths requires far fewer variables than modeling pairings since it suffices to have binary variables that connect segments together. Note that the number of pairings grows exponentially with the number of legs but the number of connections is only quadratic in the number of legs. The drawback of this approximation is that we get only 7-regular pairings and the paths that partition the remaining part of the schedule are not necessarily pairings, i.e. they are not necessarily feasible. The solution to the approximate model yields a set of 7-regular pairings that become part of the weekly crew schedule, which in turn leads to a smaller flight schedule, and we iterate the procedure on the smaller schedule by eliminating the 7-regular group and by not producing anymore 7-regular pairings. Note that not necessarily all of the 7-regular legs are assigned to the 7-regular group and therefore the 7-regular pairings obtained by the model need not partition all of the 7-regular legs. In the second iteration we obtain only 6-regular pairings and we again partition the remaining groups with paths. We repeat the procedure two more times to obtain 5-regular and 4-regular pairings. Thus we solve a series of four approximate models. Finally, we solve a weekly problem over the resulting 1-regular legs. This approach is described in more detail in Section 3.

2 Approximate Regularity Models

Now we define the four approximate models, denoted by RM for the first model, which finds only 7-regular pairings, and by $\text{MRM}_g, g = 6, 5, 4$, for the other three models. Each MRM_g model finds only g-regular pairings, and partitions the remaining legs using paths. The MRM model is a better approximation than the RM model since the paths must begin and end at the same crew base. We first define the RM model structure in Section 2.1 and then discuss the MRM_g models in Section 2.2. The cost functions are explained in Section 2.3.

2.1 The Model

As in the full regularity model, we partition the regular legs into regular groups. For the 7-regular group we require that there are 7-regular pairings that partition all the regular legs in the group. However, for the remaining g-regular groups, $4 \leq g \leq 6$ and $g = 1$, we require that paths in the segment timeline network partition the legs in the group. 'Modeling' paths requires only arc variables.

We use the same regularity variables as in the full regularity model and only a subset of the pairing variables is needed.

Pairing variables Let the pairing variable $y_p = 1, p \in S_7$, if and only if the pairing p is in the final weekly crew schedule.

We need variables that capture the connection arcs in the segment timeline network. A connection arc $(a_{i_d}, h_{j_{\tilde{d}}})$, where a_{i_d} is the node in the segment timeline network representing the arrival of segment i_d and $h_{j_{\tilde{d}}}$ is the node in the segment timeline network representing the departure of segment $j_{\tilde{d}}$, is a *g-regular connection* if it can be repeated on g consecutive days starting with day d, i.e. the segments $i_d, i_{d+1}, \ldots, i_{d+g-1}$ and $j_{\tilde{d}}, j_{\tilde{d}+1}, \ldots, j_{\tilde{d}+g-1}$ must exist in the schedule. A *g-regular path* is a path in the segment timeline network that can be repeated on g consecutive days. Every connection arc in a g-regular path is a g-regular connection.

Connection variables For each g-regular connection (a_i, h_j) from the segment timeline network, $g \in G(4,6) \cup \{1\}$, we define a binary variable z_{ijg} whose value is 1 if and only if segments i and j appear consecutively in the same g-regular path.

If leg i is in a g-regular group, $g < 7$, then there must be a connection arc that 'covers' the departure of i and there must be another connection arc that 'covers' the arrival of i, see Figure 1. We call this property, the *extendibility of a path.* Since we want the path covering i to be repeated g times, the corresponding connections must be repeated as well. If the arrival or departure station of a leg is at a crew base, then we do not require the extendibility of a path since pairings terminate and start at crew bases. Since the activity at crew bases is high due to the hub and spoke structure, there are many opportunities to patch these paths into paths that start and end at the same crew base, and eventually into pairings.

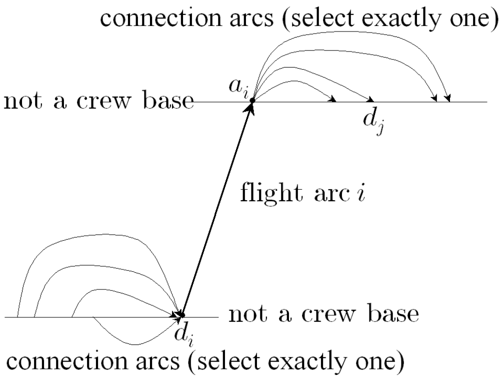
connection arcs (select exactly one)
not a crew base
a_i
d_j
flight arc i
not a crew base
d_i
connection arcs (select exactly one)

Figure 1: Extendibility of a path

Paths can be constructed from connection variables that equal 1. In addition, $x_{i,7,\mathrm{Mo}} = 1$ if and only if the 7-regular leg i is covered by a pairing $p \in S_7$ and $x_{igd} = 1, 4 \leq g < 7$ if and only if i_d is covered by a g-regular path in the segment timeline network.

Example 3. Some connection variables for the schedule given in Example 1 are

$$z_{1_{\mathrm{Mo}},6_{\mathrm{Mo}},7},\ z_{4_{\mathrm{Mo}},5_{\mathrm{Mo}},7},\ z_{5_{\mathrm{Mo}},6_{\mathrm{Mo}},7}$$
$$z_{1_{\mathrm{Mo}},6_{\mathrm{Tu}},1},\ z_{10_{\mathrm{Sa}},5_{\mathrm{Su}},1},\ z_{2_{\mathrm{Fr}},10_{\mathrm{Sa}},1}$$
$$z_{3_{\mathrm{Su}},4_{\mathrm{Mo}},6},\ z_{1_{\mathrm{Mo}},2_{\mathrm{Mo}},5},\ z_{2_{\mathrm{Mo}},3_{\mathrm{Tu}},4},\ z_{4_{\mathrm{Tu}},5_{\mathrm{We}},5}\ .$$

□

Deadheads are generally required for weekly problems. Since one of our objectives is not to use too many of them, we only consider the airline's own flights that either depart or arrive at a crew base as potential deadheads. For bigger fleets, deadheads are not needed in regular groups, but they are always required for the 1-regular group.

In addition to connection variables, the extendibility of paths can also be obtained by deadhead variables that are defined as follows, where CB is the set of all crew bases.

Deadhead connection variables For each segment i, for each $g \in G(4,6) \cup \{1\}$, and for each $cb \in CB$ such that $arr_i \neq cb$ we define a binary variable u_{ig}^{cb} which corresponds to the closest in time deadhead j from the arrival of segment i to the crew base cb such that the connection (i,j) is a g-regular connection and the connection time is within $[minSit, maxSit] \cup [minRest, maxRest]$. If such a deadhead j does not exist, the variable is not defined. Similarly we define deadhead variables v_{ig}^{cb} to cover the departure of every segment. We set $u_{ig}^{cb} = 1$ or $v_{ig}^{cb} = 1$ if and only if the deadhead is chosen.

Based on this definition, each flight has two deadheads to each crew base, one covering the departure and the other one covering the arrival. The only exception is if the arrival or departure station of the flight is at a crew base in which case there is no deadhead to this crew base.

Next we give the objective function and the constraints of the model. The objective function is

$$\min \sum_{p \in S_7} 7 \cdot \alpha_p \cdot y_p + \sum_{(i,g,d) \in Q} \beta_g x_{igd} + \sum_{i,j,g} \gamma_{ijg} z_{ijg} + \sum_{i_d,g,cb} \delta_{ig} v_{i_d,g}^{cb} + \sum_{i_d,g,cb} \delta_{ig} u_{i_d,g}^{cb}, \tag{10}$$

where the summation is only over all defined variables. The cost coefficients are discussed in Section 2.3.

Next we discuss the constraints. The generalized upper bounds constraints are

$$\sum_{(i,g,d) \in Q} x_{igd} = 1 \qquad i \in L(4,7) \tag{11}$$

and they require that each regular leg be assigned to exactly one regular group.

The 7-regular group has to be partition with 7-regular pairings which is achieved by

$$\sum_{p \in S_7 : i \in p} y_p = x_{i,7,\mathrm{Mo}} \qquad i \in L_7\,. \tag{12}$$

The remaining regular groups have to be partitioned with paths. The constraints

$$\sum_j z_{i_d j g} + \sum_{cb \in CB} u_{i_d g}^{cb} = x_{igd} \qquad (i,g,d) \in R, arr_i \notin CB \tag{13}$$

$$\sum_j z_{j i_d g} + \sum_{cb \in CB} v_{i_d g}^{cb} = x_{igd} \qquad (i,g,d) \in R, dep_i \notin CB \tag{14}$$

express the extendibility property of paths in regular groups. They reflect the fact that a path can be extended either with a connection or with a deadhead. Each 1-regular leg from L_1 must be covered by a path in the 1-regular group, hence

$$\sum_j z_{ij1} + \sum_{cb \in CB} u_{ig}^{cb} = 1 \qquad i \in L_1, arr_i \notin CB \tag{15}$$

$$\sum_j z_{ji1} + \sum_{cb \in CB} v_{ig}^{cb} = 1 \qquad i \in L_1, dep_i \notin CB\,. \tag{16}$$

We express that the 'remnants' of regular legs must be covered by paths in the 1-regular group by

$$\sum_j z_{i_d j 1} + \sum_{cb \in CB} u^{cb}_{i_d 1} = \sum_{\substack{g \in G(4,6) \\ \tilde{d}: d \in r_{ig\tilde{d}} \\ (i,g,\tilde{d}) \in Q}} x_{ig\tilde{d}} \qquad i \in L(4,7), d \in W, arr_i \notin CB, i_d \in i \tag{17}$$

$$\sum_j z_{j i_d 1} + \sum_{cb \in CB} v^{cb}_{i_d 1} = \sum_{\substack{g \in G(4,6) \\ \tilde{d}: d \in r_{ig\tilde{d}} \\ (i,g,\tilde{d}) \in Q}} x_{ig\tilde{d}} \qquad i \in L(4,7), d \in W, dep_i \notin CB, i_d \in i \,. \tag{18}$$

The regularity model, denoted by RM, has the objective function (10), constraints (11)-(18) and integrality constraints

$$x \ \text{ binary}, y \ \text{ binary}, z \geq 0, v \geq 0, u \geq 0 \,.$$

Note that given a binary x, the resulting problem for z, u and v is an assignment problem and therefore their integrality is implied by the integrality of x. Observe also that the integrality of y implies the integrality of $x_{i,7,\mathrm{Mo}}, i \in L_7$ and therefore we can relax the integrality of these variables.

Several enhancements, such as path balancedness at crew bases, crew base to crew base legs, mixing pairings and paths within a group, and extendibility of paths beyond a crew base, are presented in Klabjan (1999).

2.2 The Multicommodity Approximate Regularity Models

The connection variables equal to 1 in the RM model yield paths in the segment timeline network that start and end at crew bases, however a pairing must start and end at the same crew base. So if there is more than one crew base, we make a copy of the segment timeline network and we associate a single crew base *cb* with the network. If the RM model is applied to such an underlying network, it yields paths that start and end at the crew base *cb*. Each network requires its own set of regularity and connection variables and by linking the regularity variables, we can assign a regular leg to a unique path. This is the idea of the multicommodity regularity model, denoted by MRM_7.

For each connection variable from the RM model we have $|CB|$ connection variables, one for each crew base. We keep the same pairing and deadhead variables from RM, but the regularity variables need to be modified slightly and there are new variables that assign 'remnant' segments to a particular network. We need new 1-regular regularity variables since if a regular leg i is assigned to the network with a crew base index *cb*, a segment of i that still needs to be covered by a path can be covered in any network with an index cb'.

Regularity variables For each $i \in L_7$, let $x_{i,7,\mathrm{Mo}} = 1$ if i is assigned to the 7-regular group and 0 otherwise. For each $i \in L(4,7) \cup L_1$, for each $g \in \{1\} \cup \{4, \ldots, \min\{6, g_i\}\}$, for each $cb \in CB$, and for each day $d \in \{sd_i, sd_i + 1, \ldots , sd_i + g_i - g\}$ we define a binary regularity variable x^{cb}_{igd}. If $i \in L_7$, then the variables are defined for each $d \in W$. The cost of 1-regular regularity variables is denoted by c_1 and is discussed in Section 2.3.

We set $x^{cb}_{igd} = 1$ if and only if segment i_d is covered by a g-regular path that starts and ends at the crew base cb.

Example 4. If LAX and ORD are the only two crew bases, then the regularity variables for the schedule given in Example 1 are

$$\begin{array}{ll}
x_{i,7,\text{Mo}}, x^j_{i,6,d}, x^j_{i,5,d}, x^j_{i,4,d} & \begin{array}{l} d\in W, i=1,4,5,6,7,8,9 \\ j\in\{\text{LAX, ORD}\}\end{array} \\
x^j_{2,5,\text{Mo}}, x^j_{2,4,\text{Mo}}, x^j_{2,4,\text{Tu}} & j\in\{\text{LAX, ORD}\} \\
x^j_{3,6,\text{Su}}, x^j_{3,5,\text{Su}}, x^j_{3,5,\text{Mo}}, x^j_{3,4,\text{Su}}, x^j_{3,4,\text{Mo}}, x^j_{3,4,\text{Tu}} & j\in\{\text{LAX, ORD}\} \\
x^j_{10,1,\text{Sa}} & j\in\{\text{LAX, ORD}\} \\
x^j_{i1d} & \begin{array}{l} d\in W, i=1,4,5,6,7,8,9 \\ j\in\{\text{LAX, ORD}\}\end{array} \\
x^j_{2,1,\text{Mo}}, x^j_{2,1,\text{Fr}} & j\in\{\text{LAX, ORD}\} \\
x^j_{3,1,\text{Su}}, x^j_{3,1,\text{Mo}}, x^j_{3,1,\text{Th}}, x^j_{3,1,\text{Fr}} & j\in\{\text{LAX, ORD}\}\,.
\end{array} \qquad \square$$

The generalized upper bound constraints now read

$$x_{i,7,\text{Mo}} + \sum_{\substack{g\in G(4,6)\\ d\in W\\ cb\in CB}} x^{cb}_{igd} = 1 \qquad i\in L_7\,,$$

$$\sum_{\substack{(i,g,d)\in R\\ cb\in CB}} x^{cb}_{igd} = 1 \qquad i\in L_4\cup L_5\cup L_6\,. \tag{19}$$

In addition each 1-regular leg has to be assigned so that

$$\sum_{cb\in CB} x^{cb}_{i,1,sd_i} = 1 \qquad i\in L_1\,, \tag{20}$$

and we need constraints (12) to partition the 7-regular group with pairings. To express the extendibility property of paths in groups we have

$$\sum_j z^{cb}_{i_d jg} + u^{cb}_{i_d g} = x^{cb}_{igd} \qquad (i,g,d)\in R\cup P, arr_i\neq cb, cb\in CB \tag{21}$$

$$\sum_j z^{cb}_{j i_d g} + v^{cb}_{i_d g} = x^{cb}_{igd} \qquad (i,g,d)\in R\cup P, arr_i\neq cb, cb\in CB\,, \tag{22}$$

where $P = \{(i,1,sd_i) : i\in L_1\}$.

To complete the formulation, we need to link the regular regularity variables and the 1-regular regularity variables using the constraints

$$\sum_{\substack{g\in G(4,6)\\ \tilde{d}:d\in r_{ig\tilde{d}}\\ (i,g,\tilde{d})\in Q\\ cb\in CB}} x^{cb}_{ig\tilde{d}} = \sum_{cb\in CB} x^{cb}_{i1d} \qquad i\in L(4,7), d\in W, i_d\in i\,. \tag{23}$$

The MRM_7 model has more variables and rows than the RM model but it yields a stronger approximation. Note that as in the case with the RM model we can relax the integrality of $x_{i,7,\text{Mo}}, i \in L_7$.

2.2.1 The Generalized Multicommodity Approximate Regularity Model

The overall algorithm produces pairings in decreasing order of regularity in stages. We produce 7-regular pairings in stage 1, 6-regular pairings in stage 2, etc.

We present a model, denoted by $\text{MRM}_{g'}$, that generalizes the MRM_7 model by having only g-regular groups, $4 \leq g \leq g' < 7$, where g' is a fixed group index. The g'-regular group is partitioned with pairings and all other groups are partitioned with paths. For example, this model is used in stage 2, where 7-regular pairings have already been obtained and we want to obtain 6-regular pairings so that $g' = 6$.

In this section let $G(4, g'-1) = \{g'-1, \ldots, 4\}$ and let $L(g', 7) = \overset{7}{\underset{i=g'}{\cup}} L_i$ be the set of all regular legs that can be covered by pairings from $S_{g'}$.

We only list the parts of the model that differ from the MRM_7 model. The connection and the deadhead variables can be recycled from the MRM_7 model.

Regularity variables For each regular leg $i \in L(g', 7)$ and for each day $d \in \widetilde{D}_i = \{sd_i, sd_i + 1, \ldots, sd_i + g_i - g'\}$ we define a g'-regularity binary variable x_{id}. If $i \in L_7$, then the variables are defined for each $d \in W$ and in this case $\widetilde{D}_i = W$. $x_{id} = 1$ if and only if the segment i_d is covered by a g'-regular pairing. For each leg $i \in L(4, 7) \cup L_1$, for each $g \in \{1\} \cup G(4, g'-1)$, for each day $d \in \{sd_i, sd_i + 1, \ldots, sd_i + g_i - g\}$, and for each $cb \in CB$ we define a binary variable x^{cb}_{igd}. If $i \in L_7$, then the variables are defined for each $d \in W$.

Example 5. If LAX and ORD are the only two crew bases and $g' = 5$, then the regularity variables for the schedule given in Example 1 are

$$\begin{array}{ll}
x_{id}, x^j_{i,4,d} & d \in W, i = 1,4,5,6, j \in \{\text{LAX, ORD}\} \\
x_{2,\text{Mo}}, x^j_{2,4,\text{Mo}}, x^j_{2,4,\text{Tu}} & j \in \{\text{LAX, ORD}\} \\
x_{3,\text{Su}}, x_{3,\text{Mo}}, x^j_{3,4,\text{Su}}, x^j_{3,4,\text{Mo}}, x^j_{3,4,\text{Tu}} & j \in \{\text{LAX, ORD}\} \\
x^j_{7,1,\text{Sa}} & j \in \{\text{LAX, ORD}\} \\
x^j_{i1d} & d \in W, i = 1,4,5,6, j \in \{\text{LAX, ORD}\} \\
x^j_{2,1,\text{Mo}}, x^j_{2,1,\text{Fr}} & j \in \{\text{LAX, ORD}\} \\
x^j_{3,1,\text{Su}}, x^j_{3,1,\text{Mo}}, x^j_{3,1,\text{Th}}, x^j_{3,1,\text{Fr}} & j \in \{\text{LAX, ORD}\}\ . \qquad \Box
\end{array}$$

Pairing variables Let $y_p = 1$ if pairing $p \in S_{g'}$ is selected and 0 otherwise.

The constraints (19) are used for each $i \in L(4,7) - L(g', 7)$ and all of the constraints (20) are used. The generalized upper bound constraints

$$\sum_{d \in \widetilde{D}_i} x_{id} + \sum_{\substack{(i,g,d) \in R \\ cb \in CB}} x^{cb}_{igd} = 1 \qquad\qquad i \in L(g', 7)$$

are added for regular legs in $L(g',7)$. Since the g'-regular group has to be partitioned with pairings, we add

$$\sum_{p\in S_{g'}:i_d\in p} y_p = x_{id} \qquad i\in L(g',7), d\in \widetilde{D}_i\,.$$

We need (21) and (22) to express the extendibility property of paths. For each $i \in L(4,7) - L(g',7)$ constraints (23) are added and for $i\in L(g',7)$ we need

$$\sum_{\substack{\tilde{d}:d\in r_{ig'\tilde{d}}\\ \tilde{d}\in\widetilde{D}_i}} x_{i\tilde{d}} + \sum_{\substack{g\in G(4,g'-1)\\ \tilde{d}:d\in r_{ig\tilde{d}}\\ (i,g,\tilde{d})\in Q\\ cb\in CB}} x^{cb}_{ig\tilde{d}} = \sum_{cb\in CB} x^{cb}_{i1d} \qquad\qquad i\in L(g',7), d\in W, i_d\in i\,.$$

Note that the integrality of pairing variables implies the integrality of the g'-regularity variables x_{id}, however the remaining regularity variables are required to be binary.

2.3 Cost Function

2.3.1 Cost of Regularity and Pairing Variables

The cost of regularity variables depends only on the group index since there is no reason for prefering to assign one leg rather than another leg to a g-regular group. Because one of the objectives is to maximize regularity, if $g<\tilde{g}$, then $\beta_g>\beta_{\tilde{g}}$. However, if the absolute gap between β_g and $\beta_{\tilde{g}}$ is too large, then the objective may emphasize regularity too much. On the other hand, if the gaps are small enough, then the order of magnitude should not matter because of the generalized upper bound constraints (11). We use the values $\beta_7=1,\beta_6=2,\beta_5=3,\beta_4=4$.

Regarding the pairing cost, since a g-regular pairing is repeated g times a week, it contributes g times its pay-and-credit toward the weekly pay-and-credit. The trade-off between excess cost and regularity can be controlled by varying the relative values of the coefficients β_g.

2.3.2 Cost of Connection Variables

The sum of the cost of connection variables along a path must approximate the pay-and-credit. Let i and j be two segments such that the connection variable z_{ijg} is defined. Let ct_{ij} be the connection time. We deal separately with the cost of sit and overnight connections.

Sit Connections

The cost of a connection variable should reflect that large connection times within a pairing increase cost. In addition the cost should be a continuous function of ct_{ij}, except at plane turn connections.

For sit connections we can approximate the 'pay-and-credit' by $fd{\cdot}ct_{ij}$ since the connection time counts toward the duty elapsed time and it is weighted by fd, see (1). It is very likely that short connections do not substantially contribute toward the pay-and-credit and hence we would like to encourage them. Plane turn connections are particularly appealing.

Connections with time ct_{ij} shorter than a fixed value, denoted by $ShortTime$, are treated as short connections and are given a bonus. Let $TurnCost$ be the fixed cost of turn connections and let $ShortCost$ be the fixed cost of short connections. The contribution of a g-regular path toward the weekly pay-and-credit is g times its one day pay-and-credit and hence we weight the cost by a factor of g. The cost function for sit connections

$$\gamma_{ijg} = g \cdot \begin{cases} TurnCost & ct_{ij} < minSit, \\ ShortCost & ct_{ij} \le ShortTime, \\ fd \cdot (ct_{ij} - ShortTime) + ShortCost & ct_{ij} \le maxSit. \end{cases}$$

is shown in Figure 2.

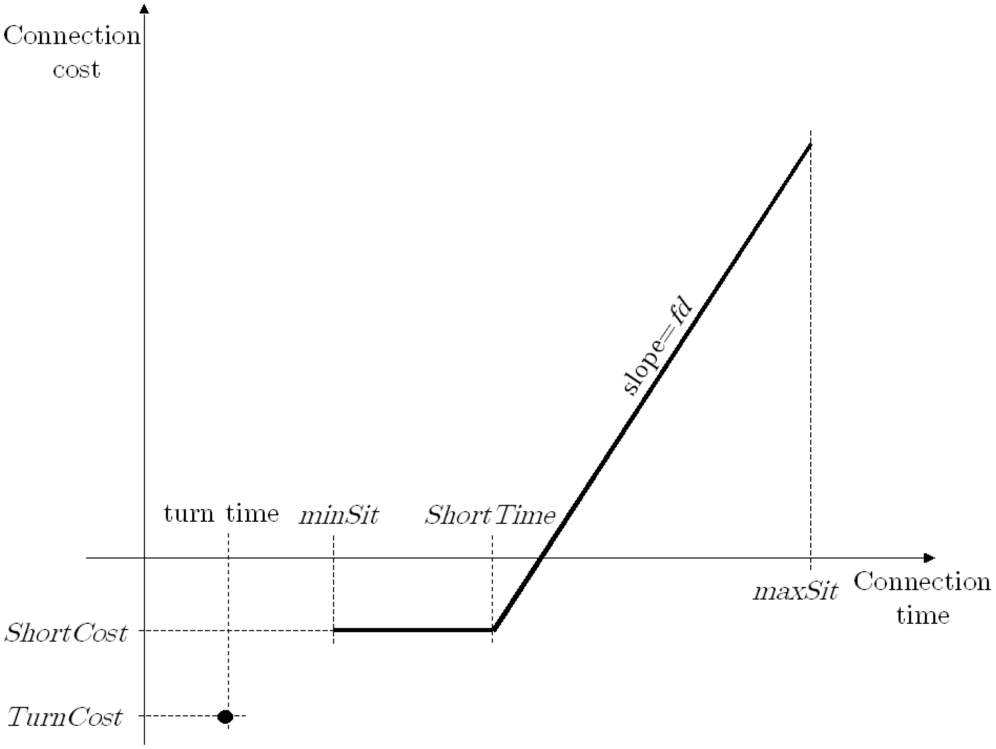

Figure 2: Connection cost curve for sit connections

Short connections at crew bases are needed to produce a diversity of pairings, which is especially needed later for the g-regular pairings with $g < 7$. We achieve this with the connection cost. We have found empirically that if all connection costs are positive, the model does not select many sit connections at crew bases for later stages. Therefore we assign negative values $TurnCost = -50$ and $ShortCost = -30$ for sit connections at crew bases and positive values $TurnCost = 15$ and $ShortCost = 30$ for connections at other stations.

Overnight Connections

We approximate the pay-and-credit contribution of overnight connections with the time away from base factor by weighting the connection time with fp, see (2). Like short sit connections, short overnight connections get a bonus. If the connection time is larger than 24 hours, then it is very likely to produce high pay-and-credit and hence we weight such connections with a larger factor. Since the span of the rest time is much larger than the range of sit connection times, we split the rest period into four subintervals. Specifically, if the connection time is short, we assign a constant cost and in the other three subintervals the slopes are $fp, k_3 \cdot fp, k_4 \cdot fp$, where $k_4 > k_3 > 1$ are parameters. Let fbp, sbp, tbp be the first, second, and third break point time, respectively, and let $FirstCost$ be the fixed cost assigned to connections in the first interval. One component of the connection cost function is

$$\widehat{\gamma}_{ij} = \begin{cases} FirstCost & minRest \leq ct_{ij} \leq fbp, \\ fp \cdot (ct_{ij} - fbp) + FirstCost & ct_{ij} \leq sbp, \\ fp \cdot (k_3 ct_{ij} + (1-k_3)sbp - fbp) \\ \qquad + FirstCost & ct_{ij} \leq tbp, \\ fp \cdot (k_4 ct_{ij} + (k_3 - k_4)tbp \\ \qquad + (1-k_3)sbp - fbp) + FirstCost & tbp \leq ct_{ij} \leq maxRest \end{cases}$$

and is shown in Figure 3 for $k_3 = 2$ and $k_4 = 5$.

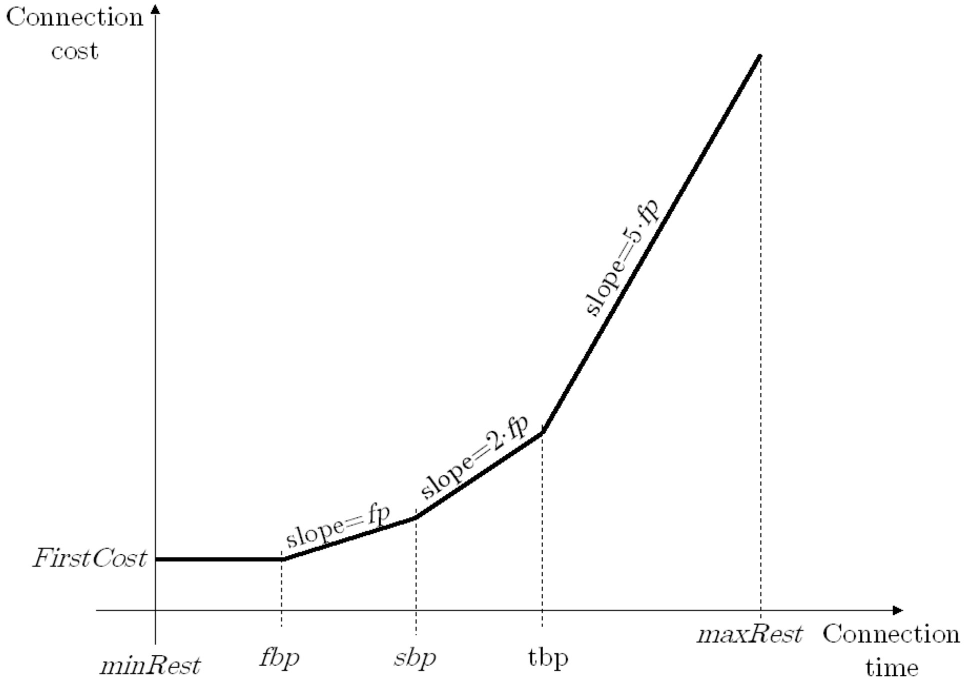

Figure 3: The component $\widehat{c}$ of the overnight connection cost

We also incorporate some aspects of duty feasibility into the overnight connection cost.

If the flight time of segment i is ‘long’, then it is preferable for segment i to be followed by an overnight connection. On the other hand, since duties with a short flying time usually produce high pay-and-credit, if the flying time is ‘short’, we prefer the next connection to be a sit connection. In this case, we penalize the overnight rest where the penalty is a decreasing function of the flying time.

If the local departure time of a segment is early in the morning, then we prefer the next connection to be a sit connection rather than an overnight connection since a natural span of duties covers the morning and afternoon. Similarly, if the local arrival time of a segment is late in the evening, we prefer to end a duty. We also create penalties to deal with these situations. Let $[bmw, emw]$ be the morning time window, let $[bew, eew]$ be the evening time window, let $ke < kd < 1$ be two constants, and let $FlightPenalty$ be a constant such that $FlightPenalty > max_{i \in L(4,7) \cup L_1} fl_i$. Then

$$pf_i = \begin{cases} fp \cdot (FlightPenalty - fl_i) & bmw \leq dt_i \leq emw, \\ ke \cdot fp \cdot (FlightPenalty - fl_i) & bew \leq at_i \leq eew, \\ kd \cdot fp \cdot (FlightPenalty - fl_i) & \text{otherwise.} \end{cases}$$

Finally, we combine the two components of the overnight connection cost to obtain

$$\gamma_{ijg} = g \cdot (pf_i + \widehat{\gamma}_{ij}) \, .$$

2.3.3 Cost of Deadhead Connection Variables

Let j be the deadhead departing from the arrival station of segment i and going to a crew base that has the shortest connection time from i, i.e. j defines u_{ig}^{cb}. The only contribution of a deadhead toward the cost of a pairing is extra connection time since deadheading does not count toward flying time. However, if we took only the pay-and-credit contribution as the cost of a deadhead connection, then we would get too many deadheads. This is undesirable since deadheads decrease revenue. We assume that the extra cost is proportional to the flight time of the deadhead, which is the case for the flight fare. Let $DhdFactor$ be the proportionality factor and let $ct_{ij} = dt_j - at_i$. We assign the cost

$$\delta_{ig} = g \cdot (q \cdot (ct_{ij} + fl_j) + fl_j \cdot DhdFactor) \, ,$$

where $q = fd$ if $ct_{ij} \leq maxSit$ and $q = fp$ otherwise. The cost of v_{ig}^{cb} is obtained in the same way.

3 Solution Methodology

3.1 General Algorithm

The overall algorithm, which is given in Algorithm 1, produces pairings in decreasing order of regularity in stages. We produce 7-regular pairings in stage 1 ($g = 7$), and then k-regular pairings in stage $8 - k$ for $k = 6, 5$ and 4. Next we describe each step in detail.

Algorithm 1: Crew Scheduling with Regularity

```
    for $g = 7$ to 4 do
1:    Generate a subset $\bar{S}_g$ of random $g$-regular pairings from $S_g$.
      if $g = 7$ then
2:      Form the regularity model RM using only pairings from $\bar{S}_g$.
      else
3:      Form the multicommodity regularity model $\text{MRM}_g$ using only pairings from $\bar{S}_g$.
      end if
4:    Solve the LP relaxation of the model.
5:    Select a subset of columns for the IP.
6:    Solve the IP using only the selected columns.
7:    Solve the crew scheduling problem on the legs assigned to the $g$-regular group and let
      $\hat{S}_g$ be the $g$-regular pairings in the solution.
8:    Select a subset of low FTC pairings from $\hat{S}_g$ to be part of the final weekly crew schedule.
      Reduce the flight schedule.
    end for
9: Find 1-regular pairings that partition the remaining schedule.
```

Step 1, pairing generation: In stage 1 we generate approximately 10 to 20 million random 7-regular pairings, see Klabjan et al. (1999). Since we do not need to cover all of the 7-regular legs by pairings in stage 1, it might suffice to consider only very low FTC 7-regular pairings in stage 1. But then we might have to use very high FTC pairings in later stages. As a compromise, we generate a substantial number of low FTC 7-regular pairings and some with moderate FTC. Specifically, the random generation is carried out twice by first generating pairings with FTC less than 0.5% and then generating pairings with FTC less than 8%. About a third of the low FTC pairings are generated. The set $\bar{S}_7$ of pairings that form the input to RM consists of all the randomly generated pairings.

In the remaining stages with $g < 7$, we proceed as follows. First we generate a segment timeline network. For each regular leg i in the current schedule with $g \leq g_i < 7$ we add segments $i_{sd_i}, \ldots, i_{sd_i+g_i-g}$ to the segment timeline network. In addition, for each 7-regular leg i that is still in the schedule and for each $d \in W$, we add i_d to the segment timeline network . Even though the schedule is substantially reduced in stage 1, the total number of pairings for the remaining stages might still be too large. If this is the case, we generate approximately 10 million random pairings in the segment timeline network which form the set of g-regular pairings $\bar{S}_g$.

Steps 2 and 3, model formation: The weaker model RM is used in stage 1 to make the model computationally tractable. In the remaining stages we use MRM_g since in stage 1 we substantially reduce the flight schedule. The models consider only pairings from $\bar{S}_g$.

Step 4, LP solving: After creating the input pairings $\bar{S}_g$ and the formulation, we first solve the LP relaxation of the model. These LP relaxations have 10,000 to 30,000 rows and up to 30 million columns. We solve the LP relaxations with a parallel primal-dual algorithm, Klabjan, Johnson and Nemhauser (2000). Linear relaxations of the MRM_g models are highly primal degenerate and hence perturbation is needed.

Step 5, selecting columns for IP: A subset of low reduced cost columns is selected for the IP using a randomized greedy approach based on the reduced cost. First we select all pairings with reduced cost below a given number $K > 0$, Klabjan et al. (1999). Since $K > 0$, we choose all of the basic pairings. If rc is the reduced cost of a pairing, then a pairing whose reduced cost is above K is selected for the IP with probability $\exp(-\tau \cdot rc^2)$, where τ is computed in such a way that the average number of selected pairings is approximately 100,000. Note that the actual number is a function of the number of legs in the problem and for smaller problems it is possible to select all of the pairings. Finally connection and deadhead variables with the best reduced cost are selected and all of the regularity variables are selected. The number of variables in IP is approximately 200,000 but the actual number depends on the number of legs in the problem.

Step 6, IP solving: Since the objective of a model is to obtain some 'good' g-regular parings, we relax the models by allowing fractional values for all but the pairing variables. Note that the integrality of pairing variables implies that all the variables x_{igd} are binary. The optimization of the relaxed problem is aborted after the first integer solution is found. We observed that fixing variables with value 1 at the root node LP relaxation does not produce good solutions.

Step 7, solving the problem on the legs in the g-regular group: Denote by $\hat{L}_g$ the set of all g-regular legs that are assigned to the g-regular group and by $\tilde{S}_g$ the set of g-regular pairings given by the solution to the model. Since the IP solution is found only on a subset of pairings from S_g and we abort the IP optimization after the first integer solution is obtained, the pairings from $\tilde{S}_g$ are unlikely to yield an optimal solution to the crew scheduling problem on the g-regular legs from $\hat{L}_g$. Therefore the crew scheduling problem on the g-regular legs from $\hat{L}_g$ is solved next, giving the set of g-regular pairings $\hat{S}_g$. This problem is solved by using the approach from Klabjan et al. (1999).

Step 8, reducing the flight schedule: Since models tend to select many g-regular pairings, some of them are 'kept active' for later stages, i.e. they are not put in the weekly crew schedule. We select a subset of low FTC pairings from $\hat{S}_g$ and these pairings become part of the final weekly crew schedule. Specifically, the schedule is updated by removing all segments $i_d, i_{d+1}, \ldots, i_{d+g-1}$ with $x_{igd} = 1$ and i_d covered by a selected pairing.

Step 9, finding 1-regular pairings on the remaining schedule: Finally, we complete the weekly crew schedule by solving the weekly problem on the remaining schedule with the solution methodology presented in Klabjan et al. (1999). Here, and only here, because of the sparsity of the schedule, it is necessary to use deadheads in pairings. Note that our use of deadhead variables in the approximate regularity models is only with respect to paths and not pairings.

3.2 Modifications

We give modifications to the algorithm that are geared toward reducing the computational time but are data dependant.

In our data sets, most legs are 7-regular, there are no 4-regular legs and all the 6- and 5-regular legs are operated every weekday. Since there are no 4-regular legs, we reduce the size of the models by considering $G(5,6) = \{5,6\}$ instead of $G(4,6)$ and by producing only 7-,6-,5- and 1-regular pairings. Since the pairings in $S_g, g \in G(5,6)$, contain only Sunday, Monday, and Tuesday segments of 5- and 6-regular legs and some segments from 7-regular legs, to make the models even smaller, we set $W = \{\text{Sa, Su, Mo, Tu, We}\}$. Therefore only segments from the 7-regular legs flown on days in W are considered in 5- and 6-regular pairings. This restriction on W substantially reduces the number of rows, and the number of regularity and connection variables in the models.

If the segment timeline network on regular legs in stage 3 is sparse, there are not many opportunities to recover infeasible paths, i.e. we cannot patch together paths into pairings, and therefore the stage 3 solution has high FTC. To overcome this difficulty, from the stage 1 and stage 2 solutions, we add to the weekly crew schedule, in Step 8, only pairings with 0 pay-and-credit. However, if the number of 0 pay-and-credit pairings is large, before solving stage 3, we remove from the weekly crew schedule all pairings that 'go through' low activity stations in the schedule for stage 3 and we add segments covered by these pairings back to the flight schedule, i.e. we create an extra operation in Step 8 in the case $g = 6$.

Even with the modification just given, in stage 3 we were unable to find good integer solutions to the model MRM$_5$, so we used a different heuristic. We find 5- and 1-regular pairings independently by removing the 1-regular part from the stage 3 model. The resulting, much easier, problem that yields only 5-regular pairings, is a set partitioning problem whose rows correspond to regular legs and columns to 5-regular pairings. The problem is solved using the daily crew scheduling algorithm given in Klabjan et al. (1999). To gain some flexibility in pairings, we let W be the set of all days in a week. Due to this approximation of the model in stage 3, we can no longer claim that there are paths partitioning the segments in the 1-regular group. Experiments have shown that the problem on the 1-regular group still has good solutions.

Algorithm 1 is designed for fleets having more than 200 regular legs. For smaller fleets in Step 1 all of the pairings in $S_g, 5 \leq g \leq 7$, are generated and considered and therefore in this case $\bar{S}_g = S_g$. In addition, MRM$_7$ is used in stage 1. The distinction between small and large fleets, which is 200 legs for our data, depends primarily on the number of pairings, which is related to the pairing feasibility parameters, and on the underlying computing environment.

4 Computational Results

All computational experiments are performed on clusters of machines. Two clusters are used, the first consisting of 16 200MHz Quad Pentium Pros and the second comprised of 48 300MHz Dual Pentium IIs, resulting in 160 processors available for parallel program execution. All machines are linked via 100 MB point-to-point Fast Ethernet switched via a Cisco 5500 network switch. Each machine with a Quad Pentium has 256MBytes of main memory whereas the remaining 48 nodes have 512MBytes of main memory per machine.

4.1 Results

The input data given in Table 1 consists of three fleets, a small one fs, a medium one fm, and a large one fl. The "Flight time" column gives the flying time in the weekly horizon. These fleets have 3 to 8 crew bases.

Name	$\vert L_7\vert$	$\vert L_6\vert$	$\vert L_5\vert$	$\vert L_1\vert$	Total
fs	90	24	5	0	119
fm	273	64	5	38	380
fl	331	104	15	42	492

Table 1: Input data

We implemented two different approaches, in one the stage 2 problem is solved and in the other one it is not. The approach denoted by 'csr2' follows the flow described in Section 3, i.e. all 3 stages are solved. The second approach, denoted by 'csr1' eliminates the computationally intensive stage 2, i.e. 6-regular pairings are not considered. Stage 1 is identical in both approaches. In 'csr1' we then solve stage 3 with the heuristic described in Section 3.2 and at the end we perform Step 9 of Algorithm 1. Note that in our implementation of 'csr1' we used a larger stage 3 than in 'csr2'.

The solution qualities of the different approaches are shown in Table 2. The 'd/e' methodology refers to the daily/exceptions approach used in current practice by most major airlines, see e.g. Vance et al. (1997). The sizes of the daily problems range from 300 to 1,600 legs, whereas the exceptions problems have 100 to 2000 segments. The measure of regularity, given in the last column, is the number of segments in the weekly crew schedule that are covered by g-regular pairings for $1 \leq g \leq 3$, the larger the number the lower the regularity. Our approach produces solutions that are better in all three aspects: FTC, number of deadheads and regularity. On average, the regularity is twice as good. The table also shows the trade-off between regularity and FTC. The solutions corresponding to the 'csr2' approach are less regular but have lower FTC. Using the methodology from Section 3, the trade-off can be easily controlled with either the cost β of the regularity variables or the number of pairings from stage 1 and stage 2 that are 'kept active' for stage 3. Even though Table 2 suggests that 'csr1' produces more regular solutions than 'csr2', this is not necessarily the case in general.

Table 3 gives further insight into the structure of solutions. Note that the 'csr1' approach does not yield any 6-regular pairings since it skips stage 2. Our solutions have a majority of their flying time in the 5- and 7-regular pairings whereas the daily/exceptions solutions have many 6-regular pairings and also have a substantial number of 4-regular pairings. Both of our approaches give many more 7-regular pairings and have fewer 1-regular pairings than the standard methodology. The result that for fs 'csr2' uses no 6-regular pairings is purely coincidental.

In Table 4 we give details on the 'csr2' approach for the fl fleet. The stage 2 problem has a large number of rows, leading to a high execution time for both the linear and the integer program. The "(IP-LP)/LP" row reports the gap between the value of the LP relaxation and the value of the obtained integer solution. The "Num. nodes" row reports the number of evaluated nodes in the branch-and-bound tree before stopping with the first integer solution.

Name	Meth.	FTC	Dhds	Measure of regularity
fs	d/e	9.2%	28	276
	csr1	7.0%	12	120
	csr2	5.6%	10	127
fm	d/e	4.2%	14	622
	csr1	3.8%	10	318
	csr2	2.5%	14	390
fl	d/e	2.27%	25	1365
	csr1	2.20%	10	520
	csr2	1.96%	8	567

Table 2: Solution quality

Name	Meth.	7-regular fl. time	6-regular fl. time	5-regular fl. time	4-regular fl. time	1-regular fl. time
fs	d/e	34%	14%	20%	14%	18%
	csr1	40%	0%	45%	0%	15%
	csr2	38%	0%	46%	0%	16%
fm	d/e	31%	26%	18%	7%	18%
	csr1	51%	0%	35%	0	14%
	csr2	31%	14%	38%	0	17%
fl	d/e	5%	26%	15%	20%	34%
	csr1	33%	0%	49%	0	18%
	csr2	20%	15%	46%	0	19%

Table 3: Flying time distribution

Integer programs are solved sequentially except that strong branching in stage 3 is performed in parallel, see Klabjan et al. (1999). The “Num. regular legs” and the ”FTC” rows list the number of regular legs in the group partitioned by pairings and the FTC of pairings in the group, respectively. The regular legs in the group are given by the IP solution to the model, i.e. before taking out high pay-and-credit pairings for later stages. The ”Best FTC” row shows the FTC of the best solution we were able to obtain on the selected regular legs. The comparison of the two ‘FTC rows’ reveals that the stage 1 solution yields pairings that are quite far from optimal. In contrast, the stage 2 solution is optimal on regular legs in the 6-regular group since there are substantially fewer pairings on the starting days of 6-regular legs that are assigned to the 6-regular group.

The IP problems are solved with the commercial branch-and-bound mixed integer programming solver CPLEX, CPLEX Optimization (1997). A steepest edge dual simplex algorithm is selected as the LP solver and the strong branching rule on variable dichotomy is used as a branching strategy. The execution times are reported in Table 5. The time for ‘csr1’ is relatively moderate while solving the stage 2 problem in the ‘csr2’ approach is extremely time

	Stage 1		Stage 2		Stage 3	
	LP	IP	LP	IP	LP	IP
Num. rows	11,316		28,447		293	
Num. cols	$17 \cdot 10^6$	200,036	$10 \cdot 10^6$	201,376	$92 \cdot 10^6$	86,075
(IP-LP)/LP	-	15%	-	14%	-	56%
Num. nodes	-	22	-	18	-	57
Exe. time (hrs)	2	7	17	14	0.1	1
Num. processors	8	1	16	1	8	40
Num. regular legs	-	235	-	223	-	293
FTC	-	2.09%	-	2.9%	-	1.46%
Best FTC	-	0.85%	-	2.9%	-	1.46%

Table 4: Solution details for the csr2 approach on the fl fleet

consuming. Given the solution qualities, the ‘csr1’ approach appears to be more practical. Its execution times are comparable to the execution times of daily/exceptions approaches, Vance et al. (1997).

Name	Meth.	Stage 1	Stage 2	Stage 3	Weekly	TOTAL
fs	csr1	0.2	0	0	0.2	0.4
	csr2	0.2	0.2	0.2	0.2	0.8
fm	csr1	8	0	1	1	10
	csr2	8	28	2	1	39
fl	csr1	12	0	2	2	16
	csr2	12	31	2	2	47

Table 5: Execution time in hours

5 Acknowledgments

This work was supported by NSF grant DMI-9700285 and United Airlines. Intel Corporation funded the parallel computing environment and ILOG provided the linear programming solver used in computational experiments.

We are especially grateful to an unusually thorough and meticulous anonymous referee whose suggestion greatly helped to improve the exposition.

References

Barnhart, C., Johnson, E., Nemhauser, G. and Vance, P. 1999. Crew Scheduling. In *Handbook of Transportation Science*. R. W. Hall (editor). Kluwer Scientific Publishers, 493–521.

CPLEX Optimization 1997. *Using the CPLEX Callable Library*, 5.0 edn, ILOG Inc.

Klabjan, D. 1999. *Topics in Airline Crew Scheduling and Large Scale Optimization*. Ph.D. Dissertation, Georgia Institute of Technology.

Klabjan, D., Johnson, E. and Nemhauser, G. 2000. A Parallel Primal-Dual Algorithm. *Operations Research Letters*, **27**, 47–55.

Klabjan, D., Johnson, E., Nemhauser, G., Gelman, E. and Ramaswamy, S. 1999. Solving Large Airline Crew Scheduling Problems: Random Pairing Generation and Strong Branching, *Technical Report TLI/LEC-99-11*, Georgia Institute of Technology. To appear in *Computational Optimization and Applications*.

Vance, P., Atamtürk, A., Barnhart, C., Gelman, E., Johnson, E., Krishna, A., Mahidhara, D., Nemhauser, G. and Rebello, R. 1997. A Heuristic Branch-and-Price Approach for the Airline Crew Pairing Problem, *Technical Report LEC-97-06*, Georgia Institute of Technology.